ตัวอย่างการใช้อะเรย์ในดีเบส

ยืน ภู่วรวรรณ

ข้อจำกัดของดีเบสมีหลายอย่างที่บางครั้งทำให้ผู้ใช้รู้สึกอึดอัดใจ เช่น จำนวนโพรซีเดอร์ไฟล์ มีจำกัดได้ไม่เกิน 32 โพรซีเดอร์ใน 1 ไฟล์ จำนวนอินเด็กซ์ที่ใช้ร่วมกับ dbf มีจำกัดได้ 8 ไฟล์ และที่สำคัญอย่างหนึ่งคือตัวแปรที่ใช้ในโปรแกรมจะมีได้ไม่เกิน 256 ตัว และดีเบสไม่มีอะเรย์ให้ใช้ ซึ่งแตกต่างกับภาษาอื่นทั้งสิ้น เช่นในเบสิกเรามี DIM A(100) ภาษาฟอร์แทรนเรามี DIMENSION A(10,10) ปาสคาลหรือซีก็มีอะเรย์ และเมื่อดีเบสสร้างมาไว้ให้เราแค่ 256 ตัวแปร เราก็จะต้องใช้ให้จำกัดอยู่ใน 256 ตัวแปรนี้ การใช้เป็นตัวแปรอะเรย์ทำได้เช่น

A&X

โดยให้ &X เป็นแมคโคร ซึ่งมีค่าตัวแปร X แปรเปลี่ยนไปเพื่อให้ตัวแปรเป็นA1,A2... ซึ่งในกรณีนี้ก็เหมือนเราใช้ตัวแปรปกตินั่นเอง

ข้อมูลแบบอะเรย์

ในที่นี้จะลองสมมติดูว่า ถ้าเรามีตารางอะเรย์ที่เกี่ยวกับรหัสสินค้าและราคาสินค้า ซึ่งมีรูปแบบดังนี้

รหัสสินค้า	ราคา
101	275.03
102	68.07
103	214.17
.	.
.	.
.	.

ข้อมูลดังกล่าวจึงมีลักษณะเป็นอะเรย์ซึ่งความจริงแล้วเราอาจเก็บในรูปไฟล์ก็ได้ แต่ในที่นี้เราจะลองเก็บแบบตัวแปรอะเรย์ดู โดยการตั้งตัวแปรเพื่อเก็บสตริงของตัวเลข เช่น X(1)=5 X(2)=8 X(3)=6 X(4)=9 เราจะเก็บไว้เป็น X ="5869"

คราวนี้ถ้าเราต้องการรู้ค่า X(3) เราก็หาได้จาก VAL (SUBSTR (X,3,1)) ซึ่งด้วยวิธีนี้เราจะเก็บตัวแปรอะเรย์ได้เหมือนกับภาษาอื่น ๆ

ลองดูตัวอย่าง

ตัวอย่างที่จะกล่าวถึงนี้เราจะเก็บรหัสสินค้าและราคาสินค้าไว้เป็นอะเรย์ดังที่แสดงในส่วนข้อมูลส่วนที่ 1

โดยให้ 3 ตัวแรกเป็นรหัสและ 5 ตัวต่อมาคือราคา จะสลับกันเช่นนี้เรื่อยไป ดังนั้นรหัส 101 มีราคา 275.03 รหัส 102 มีราคา 68.07 การเก็บข้อมูลเช่นนี้จะเก็บต่อกันไปเป็นตัวแปรสตริง 1 ตัว หากไม่พอก็ต่อตัวแปรสตริงตัวที่ 2 และ 3 เรื่อยไป อย่างไรก็ตามเราสามารถเขียนโปรแกรมปรับปรุงหรือเพิ่มข้อมูลในอะเรย์ได้เช่นกัน

เมื่อเราสร้างข้อมูลอะเรย์ได้แล้ว เราอาจเก็บไว้ในดิสก์ในลักษณะ .MEM ไฟล์ก็ได้ ทำให้ตัวแปรนี้อยู่แบบถาวรในดิสก์

ตัวอย่างโปรแกรมที่ 2 นี้ เป็นตัวนำเอาตัวแปรอะเรย์ prodtable 1 และ prodtable 2 เก็บเข้าในดิสก์

```
prodtable1="10127503102068071032141710421000105197151062200O"+;
           "10711708108127561091528111012263111152841121176З"+;
           "11309517114002501150057511601582117018261180292Я"+;
           "1190427612005326"
prodtable2="12130572122708601237141212400021125517911260002Z"+;
           "12780711128657211291825113036221131482511323671l"+;
           "13371590134052001355750013628510137628101389292O"+;
           "1396724014062350"
```

โปรแกรมที่ 1 ข้อมูลของสินค้าที่เก็บเป็นอะเรย์

```
*
*
prodtable1="10127503102068071032141710421000105197151062200O"+;
           "10711708108127561091528111012263111152841121176З"+;
           "11309517114002501150057511601582117018261180292Я"+;
           "1190427612005326"
prodtable2="12130572122708601237141212400021125517911260002Z"+;
           "12780711128657211291825113036221131482511323671l"+;
           "13371590134052001355750013628510137628101389292O"+;
           "1396724014062350"
*
*save the variable to .mem files
save to prodpric
*
*clear the memory and return to the . prompt
release all
return
```

โปรแกรมที่ 2

การใช้งานอะเรย์

การใช้งานอะเรย์จะต้องอาศัยโปรแกรมเรียกใช้ดังตัวอย่างนี้ สิ่งที่เราต้องการคือ การคีย์รหัสของสินค้า โปรแกรมจะเข้าไปหาว่ามีราคาเท่าไร ลักษณะเช่นนี้จะเห็นว่าเรามีตัวแปรอะเรย์อยู่ 2 ตัวคือ prodtable 1 และ prodtable 2 ดังนั้นจึงมีการตรวจสอบว่ารหัสที่เราคีย์เข้าไปนั้นอยู่ในตัวแปรตัวใด ถ้าไม่มีในตัวแปรก็จะรายงานข้อผิดพลาดออกมา ตัวโปรแกรมที่เราใช้เขียนได้ดังโปรแกรมที่ 3

โปรแกรมที่ 3

```
. type 6array2.prg
*array2.prg
*
*call array.prg
*     prodpric.mem
*
*Parameter passed
*           name       type  len    picture   description
*           ----       ---   ---    -------   -------------
* out    prod_nmbr      c     3     ###       Product number
* in     prod_price     N     6     ###.##    Product price
*
*Housekeeping
clear
set talk off
public is_valid,prod_price
*
*initialize memory variables with the array...
*(additive is a new feature in dbase that allows you to add
*the contents of a memory file without releaseing any
*memory variables that already exist.)
restore from prodpric additive
*
*Initialize the entry variable..
prod_nmbr="   "
*
*setup the loop for repetition..
do while .t.
  *
  *Prompt for the product number..
  @ 5,20 say "Enter the product number (return to quit)" get prod_nmbr pict "###
"
  read
  *
  *exit or call the subroutine..
  if prod_nmbr=" "
   set talk on
   release all like prod*
   return
  else
   do array
  endif
  *
  *Clear the message line and display the result..
  @ 10,25
  if is_valid
    @ 10,25 say "The price is : "+str(prod_price,8,2)+" Baht"
    * at this point, you could multiply the price by quantity
  else
    @ 10,25 say "Error in product number... please reenter"
    loop
  endif
  *
  *Housekeeping
  prod_nmbr="   "
enddo
return
*eof
```

โปรแกรมที่ 4

```
*array.prg
*
*parameter passed
*
*          name      typ   len  picture    description
*          ----      ---   ---  -------    -----------
* in       prod_nmbr C      3   ###        product number
* out      prod_price N     6   ###.##     product price
Do case
 case prod_nmbr >"100" .and. prod_nmbr <"121"
   is_valid=.t.
   array=prodtable1
 case prod_nmbr > "120" .and. prod_nmbr < "141"
   is_valid=.t.
   array=prodtable2
 otherwise
   is_valid=.f.
   prod_price=0.00
   return
endcase
*
*Search for the product number in array
pointer=1
do while prod_nmbr <> substr(array,pointer,3) .and. pointer < 160
  pointer = pointer+8
enddo
*
*Get the product price ...
prod_price=val(substr(array,pointer+3,5))/100.00
return
*eof
```

โปรแกรมของเราจะเริ่มจากการ restore ข้อมูลอะเรย์จากดิสค์ลงหน่วยความจำก่อน และที่สำคัญคือในดีเบส-ทรีมีการใช้ ADDITIVE คือนำตัวแปรเพิ่มเข้ามาโดยไม่ลบตัวแปรเดิม หลังจากนั้นจะให้เราคีย์รหัสตัวเลขสิบตัว ซึ่งเป็นข้อความ 3 ตัวอักษร เมื่ออ่านมาแล้วเราจะส่งค่ารหัสนี้ไปให้โปรแกรมย่อยชื่อ array ซึ่งหาค่าราคาแล้วส่งกลับมาให้โปรแกรมนี้พิมพ์ค่าผลลัพธ์ออกมา หากข้อมูลที่คีย์ไม่มีอยู่ในอะเรย์ก็จะรายงานข้อผิดพลาดที่เกิดขึ้น

ส่วนโปรแกรมย่อย array ที่ใช้ในการตรวจค้นหาข้อมูลในอะเรย์เขียนได้ดังโปรแกรมที่ 4

สังเกตโปรแกรมย่อย array จะรับตัวแปร prod__nmbr เข้ามาตรวจค้นข้อมูลใน array แล้วคืนค่า prod__price ออกไป หากค้นหาไม่มีหรือไม่อยู่ในขอบเขต จะให้สถานะ is_valid เป็น .F.

▼ รูปที่ 1

```
Enter the product number (return to quit) 102

The price is :     68.07 Baht
```

เมื่อพิมพ์โปรแกรมดังกล่าวแล้วเสร็จ ถ้าหาก do 6array2 จะปรากฏข้อความให้ป้อนข้อมูล ซึ่งถ้าเราป้อนข้อมูลแล้วผลจะปรากฏบนจอภาพดังรูปที่ 1

บทสรุป

ตัวอย่างที่กล่าวนี้เป็นเพียงอะเรย์แบบ dimension เดียว ถ้าหากเป็น 2 dimension หรือ 3 dimension ก็ทำได้แต่ผู้เขียนแนะนำให้ใช้ .dbf ไฟล์ร่วมด้วยจะทำให้สะดวกต่อการใช้ข้อมูลหลาย dimension ได้ ตัวอย่างเทคนิคและวิธีการนี้เป็นเพียงแนวความคิดและเทคนิคที่จะใช้กับดีเบสในกรณีที่ดีเบสไม่มีคำสั่งที่จะจัดการอะเรย์ได้โดยตรง